CS 7545: Machine Learning Theory **Fall 2021**

Lecture 10

Name: Tyler LaBonte **Professor:** Santosh S. Vempala

1 Supervised Learning

In a general setup for supervised learning, we are given some data $\mathbf{X}$ and their labels $\ell(\mathbf{X})$ for some unknown labeling function ℓ. We'd like to use an algorithm A to find a data-dependent proxy label function $H : \mathbf{X} \to \ell$. There are many ways to measure A; in particular, the distance from H to ℓ, as well as the time and sample complexity of A.

As an example, suppose the domain and labels are both booleans, and we'd like to learn the subset of the variables whose conjunction gives the label. Each literal is either present, or its negation is present, or it is not present, so there are 3^n possible conjunctions (too many to search through)! One algorithm would be, for each datapoint with label 1 eliminate either the variable (if it is a 0) or its negation (if it is a 1), and eliminate nothing if the label is a 0. However, we would need $\mathcal{O}(2^n)$ distinct 1-labeled datapoints for this algorithm to work in general, which is too many.

We will see two general models which relax the learning assumptions to enable better sample complexity.

2 PAC Learning

In the PAC, or probably approximately correct, model, our data is drawn iid from a distribution D and there is an unknown labeling function from a known hypothesis class $h \in H$. Then, A is an (ϵ, δ) PAC learning algorithm if, with probability at least $1 - \delta$, the output hypothesis g correctly classifies at least a $1 - \epsilon$ fraction of D. Such an algorithm is called efficient if the sample and time complexity is bounded by $\text{poly}(\log H, 1/\epsilon, \log(1/\delta))$.

For the conjunction example, suppose g is not ϵ-accurate. That is, $\Pr_{\boldsymbol{x} \sim D}[g(x) \neq \ell(x)] > \epsilon$. Then the probability that g is correct on m samples is bounded by $(1 - \epsilon)^m$. And, the probability that any g that has error greater than ϵ is, by a union bound, at most $(1 - \epsilon)^m 3^n \leq \delta$. Therefore, $m = \mathcal{O}(n(1/\epsilon \log 1/\epsilon + \log 1/\delta))$, so the algorithm is efficient.

3 Mistake Bound

A distribution-free model is the mistake bound model. Here, we are given datapoints one at a time, and the algorithm guesses a label z before the true label y is revealed. If $z \neq y$, we add one to the count of mistakes, and the complexity of the algorithm is the maximum number of mistakes over

all sequences of data. A mistake bound algorithm can be converted into a PAC model using the majority vote algorithm, and so the number of mistakes will be $\log H$.